1.6) List four applications you have used that most likely employed a database system to store persistent data.

- Halyk bank.

- Egov mobile.

- Kundelik (at school).

- FlyArystan.

1.7) List four significant differences between a file-processing system and a DBMS.

|  |  |
| --- | --- |
| File-processing system | DBMS |
| There is redundant data | No redundant data |
| Request processing is not as efficient | Request processing is efficient |
| Data consistency is low | Due to the normalization process, the data consistency is high. |
| Less Security | Supports more security mechanisms |

1.8) Explain the concept of physical data independence and its importance in database systems.

Physical data independence is the ability to modify the physical scheme without making it necessary to rewrite application programs. Such modifications include changing from unblocked to blocked record storage, or from sequential to random access files. Such a modification might be adding a field to a record; an application program’s view hides this change from the program.

1.9) List five responsibilities of a database-management system. For each responsibility, explain the problems that would arise if the responsibility were not dis-charged.

* If a DBMS doesn’t support defining a database; a user might invariably define non-sense as there will be no pre-defined syntax/rule.
* There will be lack of organization if the DBMS doesn’t hold the responsibility to construct the database.
* If the DBMS doesn’t hold the responsibility for manipulating the database, there will be problem arising with user trying to manipulate the database which might be querying the database or searching for information.
* If the DBMS doesn’t hold the responsibility for sharing a database; it will be hard to user to manipulate database sharing within user.
* If the DBMS doesn’t hold the responsibility for Protection; there is very much likely that the database will be exposed to various security threats.

1.11) Assume that two students are trying to register for a course in which there is only one open seat. What component of a database system prevents both students from being given that last seat?

* The **Transaction Isolation** component of the database prevents both students from being given the last seat of the course.
* In database management system always ensures the data that enters into the database must maintain ***accuracy*** and ***consistency*** of the database system.
* Therefore, when both students register for the last seat of the course, there are two probabilities presents such as:
  + *Student 1 will register for the last seat.*
  + *Student 2 will register for the last seat.*
* But there is no chance for the probability of
  + *Both students get the last seat of the course.*
  + *Not a single student gets last the seat of the course.*
* Above probabilities ensure that only one of the students would register for the last seat of the course.
* The **Transaction Isolation** concept makes this possible. It ensures that only one student's registration request is fulfilled and database maintains accuracy and consistency.
* Hence, when one student registers for the last seat of the course, then the next student would not register for the same seat or any other seat, because registration for the last seat has already been completed.

1.12) Explain the difference between two-tier and three-tier application architectures. Which is better suited for web applications? Why?

The two-tier architecture is based on Client Server architecture. The two-tier architecture is like client server application. The direct communication takes place between client and server. There is no intermediate between client and server. Because of tight coupling a 2 tier-ed application will run faster. Three-tier architecture typically comprise a presentation tier, a business or data access tier, and a data tier. Three layers in the three-tier architecture are as follows: 1) Client layer 2) Business layer 3) Data layer

Three-tier architecture is better suited for web applications. 3-tier application architecture provides a model by which developers can create flexible and reusable applications.

1.13) List two features developed in the 2000s and that help database systems handle data-analytics workloads.

• To backup data

• In some cases, to create the schema definition

• To define the storage structure and access methods

• To modify the schema and/or physical organization when necessary

• To grant authorization for data access

• To specify integrity constraints

1.14) Explain why NoSQL systems emerged in the 2000s, and briefly contrast theirfeatures with traditional database systems

The only way to increase the capacity of these databases was to upgrade the servers – processors, memory, and storage – to scale up.NoSQL databases emergedas a result of the exponential growth of the internet and the rise of web applications.

• SQLpronounced as "S-Q-L" or as "See-Quel" is primarily called RDBMS or Relational Databases whereasNoSQLis a Non-relational or Distributed Database.

• SQL databases are table based databases whereas NoSQL databases can be document based, key-value pairs, graph databases.

• SQL databases are vertically scalable while NoSQL databases are horizontally scalable.

• SQL databases have a predefined schema whereas NoSQL databases use dynamic schema for unstructured data.

• SQL requires specialized DB hardware for better performance while NoSQL uses commodity hardware.

1.15) Describe at least three tables that might be used to store information in a social-networking system such as Facebook.

* A user’s table containing users, with attributes such as account

name, real name, age, gender, location, and other profile information.

* A content table containing user provided content, such as text and

images, associated with the user who uploaded the content.

* A friend’s table recording for each user which other users are connected to that user. The kind of connection may also be recorded

in this table.

* A permissions table, recording which category of friends are allowed

to view which content uploaded by a user. For example, a user may

share some photos with family but not with all friends.